What is claimed is:

1. A parallel processing method for an inverse matrix for a shared memory type scalar parallel computer, comprising:

specifying a predetermined square block in a matrix for which an inverse matrix is to be obtained;

decomposing the matrix into upper left, left side, lower left, upper, lower, upper right, right side, and lower right blocks surrounding the square block positioned in the center;

dividing each of the decomposed blocks into the number of processors and LU decomposing the square block and the lower, right side, and lower right blocks in parallel;

updating the left side, upper, lower, and right side blocks in parallel in a recursive program, and further updating in parallel using the blocks updated in the recursive program on the upper left, lower left, upper right, and lower right blocks;

updating a predetermined square block in plural stages using one processor; and

setting the position of the square block such

that it can sequentially move on the diagonal line of the matrix, and obtaining an inverse matrix of the matrix by repeating the above mentioned steps.

2. The method according to claim 1, wherein

said shared memory type scalar parallel computer comprises a plurality of processors, plural units of cache memory provided for respective processors, plural units of shared memory, and an interconnection network for connection to be made such that the units can be communicated.

3. The method according to claim 1, wherein

said method is used to realize a Gauss-Jordan method for parallel computation for each block.

4. The method according to claim 1, wherein

a width of a division used when each block is divided for parallel computation is set such that a total amount of computation of square blocks not processed in parallel can be about 1% of entire computation from a size of a matrix from which an inverse matrix is obtained and from a number of processors available in a parallel process.

5. A program for realizing a parallel processing method for an inverse matrix for a shared memory type scalar parallel computer, comprising:

specifying a predetermined square block in a matrix for which an inverse matrix is to be obtained;

decomposing the matrix into upper left, left side, lower left, upper, lower, upper right, right side, and lower right blocks surrounding the square block positioned in the center;

dividing each of the decomposed blocks into the number of processors and LU decomposing the square block and the lower, right side, and lower right blocks in parallel;

updating the left side, upper, lower, and right side blocks in parallel in a recursive program, and further updating in parallel using the blocks updated in the recursive program on the upper left, lower left, upper right, and lower right blocks;

updating a predetermined square block in plural stages using one processor; and

setting the position of the square block such that it can sequentially move on the diagonal line

of the matrix, and obtaining an inverse matrix of the matrix by repeating the above mentioned steps.

6. The program according to claim 5, wherein said shared memory type scalar parallel computer comprises a plurality of processors, plural units of cache memory provided for respective processors, plural units of shared memory, and an interconnection network for connection to be made such that the units can be communicated.

7. The program according to claim 5, wherein said method is used to realize a Gauss-Jordan method for parallel computation for each block.

8. The program according to claim 5, wherein a width of a division used when each block is divided for parallel computation is set such that a total amount of computation of square blocks not processed in parallel can be about 1% of entire computation from a size of a matrix from which an inverse matrix is obtained and from a number of processors available in a parallel process.